Alsa−sound−mini−HOWTO

Table of Contents

Table of Contents

Table of Contents

Alsa−sound−mini−HOWTO

Valentijn Sessink `valentyn@alsa-project.org`

v2.0−pre1, 12 November 1999

Describes the installation of the ALSA sound drivers for Linux. These sound drivers can be used as a replacement for the regular sound drivers, as they are fully compatible.

---

6. Testing and using

7. Tips and Troubleshooting

---

1. Introduction

This is the ALSA Sound drivers mini−HOWTO. It gives you information about installing and using the ALSA sound drivers for your soundcard. The ALSA drivers are fully modularized sound drivers that support kerneld and kmod. They are compatible with, but surpass the possibilities of, the current OSS API. In other words: compatible, but better.

1.1 Acknowledgments

This documents contains information I got from the ALSA driver page. The structure was ripped off the SB−mini−HOWTO, mainly because it had about the structure I was looking for. Thanks to the SGML Tools package, this HOWTO is available in several formats, all generated from a common source file. Thanks to Erik Warmelink for proof reading, thanks to Alfred Munnikes for a couple of questions and helpful suggestions. Yamahata Isaku thanks for the Japanese translation, Miodrag Vallat for the translation in French. Later on, Steve Crowder did a great job by reading and editing the whole text. Thanks to Cserna Zsolt for the Hungarian translation and Marco Meloni for the Italian one. Thanks to Mohamed Ismail Mohamed−Ibrahim who sent me a document about the Trident 4DWave DX/NX soundcard with a lot of useful information, thanks to Gerard Haagh who sent me a lot of useful information and who also pointed out a few unclear sections.

Thanks to Marc−Aur`ele Darche, Piotr Ingling, Juergen Kahrs, Tim Pearce, Patrick Stoddard, Rutger de Graaf, Shuly Wintner, Jyrki Saarela, Jonas Lofwander, Kumar Sankaran and many others for useful tips and

additions.

1.2 Revision History

Version 2.0−pre1 − November 12, 1999. Updating a couple of sections to ALSA 0.4.1e., added various links.

Version 1.7 − July 29, 1999. A few fixes.

Version 1.6 − July 26, 1999. Added a section about ALSA−versions

Version 1.5 − May 21, 1999. Changed the mixer section, added a quick install section

Version 1.4 − May 18, 1999. Included the URL to the French version, changed more URLs.

Version 1.3 − May 16, 1999. Thanks to Jaroslav this HOWTO has found a home at the ALSA−project website. As a result of that, some updates in mail and web addresses.

Version 1.2 − May 11, 1999. Several updates.

Version 1.1 − March 11, 1999. Added a couple of sound cards from the new 0.3 series drivers, wrote a bit about the 2.2 series kernel.

Version 1.0 − February 8, 1999. Added a few things to the troubleshooting section, but we seem fairly complete.

Version 0.3 beta − January 20, 1999. A link on the ALSA−homepage. Ha, we're official!

Version 0.2 alpha − Mid January 1999, first .sgml−version.

Version 0.1 alpha − January 1999, first version, mostly HTML.

Still: please submit any patches in plain English, you native speakers!

There are a couple of additions that need to be added to the HOWTO now. Notably, Mohamed Ismail Mohamed−Ibrahim and Gerard Haagh wrote wonderful additions to the HOWTO, that will keep me off my regular work for some more time. So this is 2.0−pre1 and more pre's are to follow.

1.3 New versions of this document

The latest version can be found at http://www.alsa−project.org./~valentyn

Other formats (full size html, sgml, txt) are in the directory other−formats. Unfortunately, I have not succeeded in compiling a Postscript version, as the sgml2latex−script returns a bunch of errors.

Yamahata Isaku has translated a Japanese version, which will be available at the Japanese ALSA site, http://plaza21.mbn.or.jp/~momokuri/alsa/index.html

Miodrag Vallat translated a French version, which is available at http://www.freenix.fr/unix/linux/HOWTO/mini/Alsa.html.

Cserna Zsolt has translated the Hungarian version of the ALSA−HOWTO. You can find it at http://kib4.vein.hu/~zsolt/alsa.html.

Marco Meloni did an Italian version, you can get it at http://pluto.linux.it/ildp/index.html.

If you make a translation of this document into another language, let me know and I'll include a reference to it here. Ook een Nederlandse versie is welkom, ik heb zelf geen tijd om deze te schrijven. Leve de koningin!

1.4 Feedback

I rely on you, the reader, to make this HOWTO useful. If you have any suggestions, corrections or comments, please send them to me (alsa−howto@alsa−project.org), and I will try to incorporate them in the next revision.

Please note: I do not get a lot of mail about the ALSA drivers and any addition is welcome. Even a ``thank you for'' is appreciated − maybe it's not too much work to add a ``I appreciated most'' or ``this−or−that was not immediately clear to me''−section.

If you publish this document on a CD−ROM or in hardcopy form, a complimentary copy would be appreciated. Mail me for my postal address. Also consider making a donation to the Linux Documentation Project to help support free documentation for Linux. Contact the Linux HOWTO co−ordinator, Tim Bynum linux−howto@metalab.unc.edu, for more information.

1.5 Distribution Policy

---

2. NOWTO − a quick install guide

If you want sound and you want it NOW! and not after reading this HOWTO, this quick tour through the ALSA driver installation might be of help. Please note: there are a couple of differences between the ALSA versions that support 2.0 kernels and those that support 2.2 kernels.

2.1 Installing ALSA for kernels 2.2.x

You will probably want to use the ALSA 0.4.1e (or later) version if your kernel is 2.2.x. If your kernel is older, please use 0.3.0−pre4 and see below.

Just the all time ``./configure − make − make install'' stuff. Do this for **drivers, library and utilities**. You need all three because the utilities help you to unmute your card. Kernels 2.2.x need to have **general sound support** in the kernel (without choosing a specific card).

The ALSA drivers have their own devices, you can make them usinge the **./snddevices** script.

You need to load the module for your card (or use kmod) and if you want sound to be backwards compatible with the Linux kernel sound drivers (yes you want this) you need two other modules called **snd−pcm1−oss** and **snd−mixer−oss**. See the section Which module for which card to find out which module to load. After loading, you can look in /proc/asound for various information about the ALSA drivers.

2.2 Playing and recording sound

A few remarks. ALSA has it's own devices in /dev/snd, for example /dev/snd/pcmC0D1 is Card 0, Device 1. You can use the old /dev/pcmXY devices if you loaded snd−pcm1−oss for backwards compatibility. You'll also want to use /dev/mixer, so load snd−mixer−oss as well. Before you can play any sound, you need to unmute the card with ``amixer''. Type ``amixer groups'', then try something like

```
amixer set PCM 100 unmute
```

Generally you can use options ``mute'' or ``unmute'', ``capture'' or ``nocapture'' and numbers.

That's it! Now if it works, it works. If it doesn't work, you may need to actually read this HOWTO...

2.3 Installing ALSA for 2.0.x

The ALSA drivers versions 0.3.0, 0.3.1 and 0.3.2 have various problems due to the restructuring of the mixer interface. Later versions do not support kernel 2.0.x, so you definately will want to use version 0.3.0−pre4 if you have a 2.0 version kernel.

Just the all time ``./configure − make − make install'' stuff. Do this for **drivers, library and utilities**. You need all three because the utilities help you to unmute your card. Kernels 2.0.x need to have all **sound support disabled** in the kernel setup

The ALSA drivers have their own devices, you can make them usinge the **./snddevices** script.

You need to load the module for your card (or use kmod) and if you want sound to be backwards compatible with the Linux kernel sound drivers (yes you want this) you need another modules called **snd−pcm1−oss**. See the section Which module for which card to find out which module to load. After loading, you can look in /proc/asound for various information about the ALSA drivers.

2.4 Playing and recording sound

A few remarks. ALSA has it's own devices in /dev/snd, for example /dev/snd/pcmC0D1 is Card 0, Device 1. You can use the old /dev/pcmXY devices if you loaded snd−pcm1−oss for backwards compatibility. Before you can play any sound, you need to unmute the card with ``amixer''. Type ``amixer'', then try something like

```
amixer pcm 100 unmute
```

Generally you can use options ``mute'' or ``unmute'', ``rec'' or ``norec'', numbers or left:right.

That's it! Now if it works, it works. If it doesn't work, you may need to actually read this HOWTO...

---

3. Before you start

3.1 Introduction

This document tries to help you install and use the ALSA sound drivers in your Linux system. The reference system is a Slackware 4.0 distribution of Linux on an AMD/K6 computer (x86 compatible), but it should work with any other Linux distribution. I do not know if the ALSA drivers work on other platforms, according to the documentation, Alpha has been tested and proven to work. I have only x86 PC's here, so any additional information you may have would be appreciated.

It might be handy to read the Linux Sound HOWTO (see section Other HOWTO's), but that HOWTO focuses on the built−in kernel drivers.

3.2 General information about the ALSA drivers

The ALSA sound driver was originally written as a replacement for the Linux kernel sound for Gravis UltraSound (GUS) cards. As this GUS replacement proved to be a success, the author started the ALSA project for a generic driver for several sound chips, with fully modularized design.

It is compatible with the OSS/Free and OSS/Linux sound drivers (the drivers in the kernel), but has its own interface that is even better than the OSS drivers. A list of features can be found at http://www.alsa−project.org/intro.html

Please note that the ALSA drivers are still under development. Things may change over time, and some programs that rely on ALSA only work under specific versions of it. Apart from that: I think they're great. I use ALSA for 10 months now and will never go back to the dark ages of closed source sound drivers − hint ;)

The main page of the ALSA project is http://www.alsa−project.org/

3.3 Supported hardware

The ALSA drivers support only a subset of all sound cards available. As the time of writing, the following cards are supported.

- Cards with a Trident 4D Wave DX/NX chipset, thanks to Trident Microsystems who offered ALSA ``first cut'' GPL'd drivers (MIXER and PCM devices only) and documentation for their 4D Wave PCI audio chipsets. See http://www.tridentmicro.com/HTML/products%20folder/audio.htm for more information. Cards using this chipset include: Best Union Miss Melody 4DWave PCI, HIS 4DWave PCI, Warpspeed ONSpeed 4DWave PCI, AzTech PCI 64−Q3D, Addonics SV 750, CHIC True Sound 4Dwave, Shark Predator4D−PCI and Jaton SonicWave 4D.
- Gravis Ultrasound (GUS): ``PnP'', Extreme, Classic/ACE, MAX

- Cards with a GUS chipset: Dynasonic 3−D, STB Sound Rage 32, UltraSound 32−Pro (STB), ExpertColor MED3201 and others with AMD InterWave chip, notably some STB cards by Compaq
- Soundblaster: 1.0, 2.0, Pro, 16, AWE32/64, PCI64
- ESS AudioDrive ESx688
- ESS ES968 chip based cards (PnP only).
- ESS ES18xx (chipsets). Please note that I personally experienced a lot of trouble with the ESS1888. The developer of the driver for this card did his best, but to no avail.
- ESS Solo−1 ES1938 and ES1946. Only one of the two channels works, which means that recording is not possible. The author of the ES1938 code ``is aware of the problem and is currently investigating it''.
- Yamaha: OPL3−SA2, OPL3−SA3 (chipsets)
- OAK Mozart
- Schubert 32 PCI (PINE, S3 SonicVibes PCI chipset)
- Ensoniq AudioPCI ES1370/1371 PCI soundcards (Soundblaster PCI64)
- SonicVibes PCI soundcards (PINE Schubert 32 PCI)
- ForteMedia FM801 based cards (in 0.3.2)
- OPTi 82C9xx chipset based soundcards
- AD1847, AD1848 and CS4248 chipset based cards
- AZT2320 chip based soundcards (PnP only).
- Advance Logic ALS100/ALS120 based cards
- C−Media CMI8330 based cards

Then a whole lot of Crystal Semiconductors−based sound boards are supported. These chips can be found in a lot of hardware, in separate cards (some Philips PCA series) and on motherboards (e.g. IBM Aptiva, Dell computers). Boards based on the following chipsets are supported:

- 4231
- 4232
- 4232A
- 4235
- 4236B
- 4237B
- 4238B
- 4239
- 4280
- 4610
- 4612
- 4614
- 4615
- 4680

The best thing is: ALSA now supports computers without a soundcard to produce video! This is done with a dummy driver, that tricks programs like Realplayer into thinking that there is a sound card available.

A more recent list may be found inside the driver package itself, that is in doc/SOUNDCARDS

3.4 Other HOWTO's

This ALSA−sound−mini−HOWTO is just mini − although it seems to grow fast. Other HOWTO's may help you out in case this one is too terse. I will name a few things you may come across while trying to install the ALSA drivers. HOWTO's can generally, be found at mirrors of Metalab (the former Sunsite). So take a look at http://metalab.unc.edu/LDP/mirrors.html and pick out your closest mirror site. You can find HOWTO's in the directory LDP/HOWTO/. Please note: the links in this document will all be relative to

/LDP/HOWTO/mini. If you look at this document from a reasonably good mirror site, you will find the HOWTO's.

Then a note for the 2.2.x kernel series. For the 2.2.x kernel series, sound support is like any other support: it works, but it is different from what you used to do. This HOWTO (like any other HOWTO) will from version 2.0pre1 concentrate on the 2.2 series kernel, although I'll try to point out the differences.

Sound cards

Perhaps you bought a sound card already, or maybe it has been installed in your computer for ages. And now you are going to use it! Have a look at the Sound−HOWTO to see if this is all worth the trouble. (You might want to buy this new Mega−Rumble−Blaster first, then try the ALSA drivers.)

Plug and Play cards

Most modern sound cards for the Intel platform are ISA PnP cards, which is an abbreviation for "Plug and Play". This means, that the card has to be configured by the operation system. This has to be done through an initialization routine at boot time. You probably need to configure your card with the PnP−utils−package. Every recent Linux distribution includes these tools. For usage have a look at the Plug−and−Play−HOWTO

The ALSA−drivers seem to have built in their own ISA−PnP−support for a couple of sound cards. Unfortunately, as I cannot find documentation about this, I cannot tell you how it works. If anyone out there wants to try ALSA sound support while deliberately not using the ISA−PnP−tools, please drop me a line.

Loadable modules

The ALSA sound drivers are built as modules. You can find more information about modules in the Kernel−HOWTO. There is also a module−HOWTO, but that is unmaintained at the moment; take a look at the umaintained section of the Howto−HOWTO. There is a Modules−mini−HOWTO though that may be useful.

Kerneld

Another HOWTO that will be useful for some, is the Kerneld−mini−HOWTO. Kerneld is a daemon that installs and removes kernel modules as needed. (I have zero experience with it, so additional information on the topic is welcome. The ALSA driver documentation contains some information about configuration of the kerneld, this has been included in this mini−HOWTO.)

As the kernel module loader is included in kernel 2.2.x, things have changed. But as I am one of those guys that rather modprobes something than have some daemon handle it, I have no info on this.

---

4. How to install ALSA sound drivers

4.1 What you need

- a functional Linux system (e.g. the Slackware distribution), with the "Development" packages installed (i.e. gcc, make etc.)

- some knowledge about Linux (meaning you know how to use "ls", "cd", "tar" etc.)
- a root−account

The great thing is: you don't need a supported sound card anymore, as ALSA now has a dummy driver that does nothing! (No, it really does nothing, but some programs will work now that they believe there is a sound card available).

If you have a PnP card, you will also need:

- the isapnptools software package.

The INSTALL text in the driver directory suggests that for some cards, PnP support is native. I also received a suggestion from Jaroslav about this. When I get further information about this topic I will add it to this mini−HOWTO.

Please note that you should not have any sound drivers active when you want to use the ALSA drivers. If you have a kernel with sound drivers compiled in, you'll need a kernel recompilation. If you have the old "sound.o" module active, you need to deactivate it. If you use kerneld, this probably means deleting sound.o from the /lib/modules/<kernel_version>/misc directory. Newer RedHat systems have a different sound approach, with several sound modules active. You need to deactivate them all.

The 2.2 series kernel has a new approach to sound. You should include sound support here ! Yep, that's right: you add sound support to the kernel, but do not include any sound card. Then compile and install the kernel and after that, compile the ALSA−drivers.

4.2 Getting the drivers

The ALSA drivers are available from ftp://ftp.alsa−project.org/pub/ and there are mirrors at

- US: ftp://ftp.silug.org/pub/alsa
- US: ftp://ftp.eecs.umich.edu/pub/linux/alsa
- Netherlands: ftp://linux.a2000.nl/alsa
- Poland: ftp://ftp.task.gda.pl/pub/linux/misc/alsa
- Germany: ftp://ftp.tu−clausthal.de/pub/linux/alsa
- Slovakia: ftp://ftp.phacka.sk/pub/alsa
- Australia: ftp://ftp.suburbia.com.au/pub/alsa

For a fully functional ALSA−installation, you will need the driver, the libs and the utilities; e.g if you chose the A2000 mirror you would get ftp://linux.a2000.nl/alsa/driver/alsa−driver−0.4.1e.tar.gz, ftp://linux.a2000.nl/alsa/lib/alsa−lib−0.4.1d.tar.gz and ftp://linux.a2000.nl/alsa/utils/alsa−utils−0.4.1.tar.gz

4.3 ALSA versions

The ALSA drivers have come a long way. Development started during the 2.0 version kernel, then the 2.2 series showed up (with their own sound kernel).

As the 0.4 versions work perfectly for me, I think it is safe to use 0.4.1e (or newer, if you want). If you have a 2.0.x kernel, you will definately not want to use 0.3.0 or later. Instead, use alsa−driver−0.3.0−pre4, alsa−lib−0.3.0−pre4 and alsa−utils−0.3.0−pre3.

The older versions, 0.2.0−pre10p3 and older do work under 2.0.x, but I cannot get them to work under 2.2.x (probably due to the lack of interfacing with the soundcore module of the kernel).

4.4 Extracting

You extract the drivers by some reasonable command, like the all−time tar −zxf <file>. Most likely you would do that in the /usr/src directory, so you need root priviliges for this. Type ``su'' and then the root password to become root. But please note: it is unwise to use your system as the ``root'' user if it is not necessary. So:

```
cd /usr/src

tar -zxf ~/alsa-driver-0.4.1e.tar.gz

tar -zxf ~/alsa-lib-0.4.1d.tar.gz

tar -zxf ~/alsa-utils-0.4.1.tar.gz
```

Also working and more fun: find ~ −name alsa* −exec tar −zxf {} \; (Don't try this at home kids, it's just an example). Note that when downloading the drivers with Netscape, you may accidentally get unpacked drivers with a ".tgz" extension. If tar complains about the file format, you may get better results by leaving off the "z" in the tar options.

4.5 Compiling

You need the drivers before you can compile and use the libs. You need the libs before you can compile or use the utils. So let's begin:

```
cd alsa-driver-0.4.1e
```

(and for those not so experienced: try typing a <tab> (the "tab"−key) after "alsa−d". That's called *command line completion*.)

```
./configure
```

If you want to use the built−in PnP interfacing, you should use

```
./configure --with-isapnp=yes

make
```

Now you need to be 'root' to install the stuff (you probably were "root" already)

```
make install
```

If this tells you that something like ``version.h'' cannot be found, then you probably do not have a proper kernel source tree. You need a couple of files of your kernel source to be able to compile the ALSA−drivers. Unpack your favorite linux−2.x.y.tar.gz in /usr/src, and issue a `make menuconfig`. (Actually, `make symlinks` may be enough). Now compile the libraries:

```
cd ../alsa-lib-0.4.1d

./configure

make
```

```
make install
```

OK, you're getting it, the utilities:

```
cd ../alsa-utils-0.4.1

./configure

make

make install
```

Note: you can leave out the "make install" for the utilities at first. You could even leave out the whole library−making and utility−making, just to check if the driver works.

4.6 Preparing the devices

There is a script in the driver−directory that will install the ALSA−sound−devices in your /dev directory. Type

```
./snddevices
```

from the driver−directory. There should be a /dev/snd subdirectory now (test if it is there. If you are not familiar with even the "ls" command, please consider reading other HOWTO's first. You should have some basic Linux knowledge to install these drivers).

Now you're ready to insert the driver, so please turn over to the next paragraph.

---

5. Loading the driver

There are two ways to use the ALSA−sound−modules. I personally prefer using the manual method, meaning that I insert the driver at startup. The ALSA−drivers were designed as loadable/unloadable modules − for instance they do not reset the mixer after loading − so you can easily use the kerneld approach.

Please do read the section Backwards Compatibility. You need it to have sound support ``the old way''.

5.1 Inserting with modprobe

Note: If you have a PnP audio−card, you first need to set it to the right (or at least some known) IO/IRQ/DMA. See the Plug−and−Play−HOWTO. Did you configure your Plug−and−Play−soundcard? Ok, then read on please. The main part is: do a "modprobe snd−card−<soundcard>". This should do the trick. Please note that not all distributions do include /sbin in your path. If you get a "bash: modprobe: command not found", this will most likely mean that modprobe is not in your path. Try ``/sbin/modprobe snd−card−sb16'', or try to find the modprobe utility elsewhere.

The most important difficulty is with the Crystal chipsets, for these the ALSA−drivers are not auto probing. More recent information may be acquired from the INSTALL file in the driver−directory. Two examples, then a list: Gravis UltraSound (GUS) and compatibles:

```
/sbin/modprobe snd-card-gusclassic
```

For all 16−bit Soundblaster−cards (SoundBlaster 16 (PnP), SoundBlaster AWE 32 (PnP), SoundBlaster AWE 64 (PnP):

```
/sbin/modprobe snd-card-sb16
```

However, if you have a 0.3.0−pre4 package, the GUS Classic driver is called ``snd−gusclassic'' and the SoundBlaster 16 module is called ``snd−sb16'' (so, without the ``card'' part).

5.2 Which module for which card?

Please note that ALSA versions before 0.4.x sometimes had different names. You need to leave out the ``card'' part for those drivers. This is indicated by an asterisk (*).

Gravis UltraSound Extreme (*)

```
modprobe snd-card-gusextreme
```

Gravis UltraSound MAX (*)

```
modprobe snd-card-gusmax
```

ESS AudioDrive (*)

ESS AudioDrive ES−1688 and ES−688 soundcards

```
modprobe snd-card-audiodrive1688
```

ESS AudioDrive 18xx (*)

ESS AudioDrive ES−18xx based soundcards

```
modprobe snd-card-audiodrive18xx
```

Gravis UltraSound PnP (*)

Gravis UltraSound PnP, Dynasonic 3−D/Pro, STB Sound Rage 32, ExpertColor MED3201 and other soundcards based on AMD InterWave chip.

```
modprobe snd-card-interwave
```

UltraSound 32−Pro (*)

UltraSound 32−Pro (soundcard from STB used by Compaq) and other soundcards based on AMD InterWave (tm) chip with TEA6330T circuit for extended control of bass, treble and master volume

```
modprobe snd-card-interwave-stb
```

Soundblaster (*)

8−bit Soundblaster cards (SoundBlaster 1.0, SoundBlaster 2.0, SoundBlaster Pro)

```
modprobe snd-card-sb8
```

Soundblaster 16 (*)

16−bit SoundBlaster cards (SoundBlaster 16 (PnP), SoundBlaster AWE 32 (PnP), SoundBlaster AWE 64 (PnP). Please note: this module does not support the SoundBlaster VibraX16 soundcard.

```
modprobe snd-card-sb16
```

OAK Mozart (*)

```
modprobe snd-mozart
```

OPTi 82C9xx (*)

Various sound cards that use the OPTi 82C9xx chipset, like Audio 16 Pro EPC−SOUN9301 (82C930 based), ExpertColor MED−3931 v2.0 (82C931 based), ExpertMedia Sound 16 MED−1600 (82C928 based – AD1848), Mozart S601206−G (OPTI601 based – CS4231) and Sound Player S−928

```
modprobe snd-card-opti9xx
```

AD1847/48 and CS4248

```
modprobe snd-card-ad1848
```

Yamaha OPL3−SA2/SA3 soundcards (*)

Just "modprobe snd−opl3sa" will not work, this driver does not do autoprobing. See below.

S3 SonicVibes (*)

S3 SonicVibes PCI soundcards. (PINE Schubert 32 PCI)

```
modprobe snd-card-sonicvibes
```

Ensoniq/Soundblaster PCI64 (*)

Ensoniq AudioPCI ES1370/1371 PCI soundcards. (SoundBlaster PCI 64)

```
modprobe snd-card-audiopci
```

CS4231

Just ``modprobe snd−card−cs4231'' will not work, no auto−probing. See below.

CS4232/4232A

All soundcards based on CS4232/CS4232A chips. Just "modprobe snd−card−cs4232" will not work, no auto−probing. See below.

4235 and higher

All soundcards based on CS4235/CS4236/CS4236B/CS4237B/CS4238B/CS4239 chips. Just "modprobe snd−card−cs4236" will not work, no auto−probing. See below.

4610/4612/4615 and 4680

```
modprobe snd-card-cs461x
```

ESS Solo 1 (*)

ESS Solo−1, 128iPCI card (es1938, ESS−SOLO−1). Jonas Lofwander sent me a link to a document that will help you installing this card − which is, basically, nothing more than `modprobe snd-card-esssolo1` ... but http://dice.shopcenter.nu/alsa/ can be of help. If you have an IBM Thinkpad 1412 you can also refer to http://www.geocities.com/SiliconValley/Peaks/3649/1412.html, thanks to Kumar Sankaran.

Trident 4DWave DX/NX (**)

Best Union Miss Melody 4DWave PCI, HIS 4DWave PCI, Warpspeed ONSpeed 4DWave PCI, AzTech PCI 64−Q3D, Addonics SV 750, CHIC True Sound 4Dwave, Shark Predator4D−PCI, Jaton SonicWave 4D.

```
modprobe snd-card-trident
```

ForteMedia FM801

These are PCI cards based on the FM801 chip.

```
modprobe snd-card-fm801
```

(*) For ALSA version 0.3.0−pre4, you need to leave out the ``card−'' part in most (not all!) of the drivernames. So ``snd−card−sb16'' becomes ``snd−sb16'', however, ``snd−card−cs4232'' remains ``snd−card−cs4232'' (modprobe snd−cs4232 will do something, but it will not produce any sound!)

(**) In older ALSA versions this driver was called ``snd−card−trid4wave'' and ``snd−trid4wave''.

5.3 modprobe for drivers without auto−probing

If you have a non−autoprobing driver, you need to supply additional info at startup to have the driver work. More information can be found in the file INSTALL in the driver directory.

OPL3−SA2 and OPL3−SA3

According to the INSTALL file you need to supply all the information for this driver. If you initialized the card with the isapnp−tools, you can probably get info from the /etc/isapnp.conf file for the following values:

```
snd_port - control port # for OPL3-SA chip
snd_wss_port - WSS port # for OPL3-SA chip (0x530,0xe80,0xf40,0x604)
snd_midi_port - port # for MPU-401 UART (0x300,0x330), -1 = disable
snd_fm_port - FM port # for OPL3-SA chip (0x388), -1 = disable
snd_irq - IRQ # for OPL3-SA chip (5,7,9,10)
snd_dma1 - first DMA # for Yamaha OPL3-SA chip (0,1,3)
snd_dma1_size - max first DMA size in kB (4-64kB)
snd_dma2 - second DMA # for Yamaha OPL3-SA chip (0,1,3), -1 = disable
snd_dma2_size - max second DMA size in kB (4-64kB)
```

You would do a "modprobe snd−card−opl3sa snd_port=0xNNN snd_wss_port=0x530 snd_midi_port=−1 snd_fm_port=0x388 snd_irq=5 snd_dma1=0 snd_dma1_size=NN snd_dma2=1 snd_dma2_size=NN" to load this driver (without midi−support. I am still convinced that midi−support is the thing you need when you have synthesizers and stuff and want to connect them to your Linux box. Never needed Midi−support even to play midi−files.)

Note that the "NN" values need to be supplied, only I do not know what would be reasonable values. I do not know if the dma size option is really required.

If you happen to have an IBM Thinkpad with this chipset, then http://www.cirs.org/patrick/index.html might be of help.

If you use the driver from 0.3.0−pre4, then leave out the ``card−'' part in the name.

CS4231 chips

According to the INSTALL file you need to supply the main port for this card. Note that with the driver for 3235/6/7/8/9 cards, the one below, I ended up supplying all information (except DMA−size), otherwise the driver did not work. So you may as well use the whole command line to insert the driver. If you initialized the card with the isapnp−tools, you can probably get info from the /etc/isapnp.conf file for the following values:

```
snd_port - port # for CS4232 chip (PnP setup - 0x534)
snd_mpu_port - port # for MPU-401 UART (PnP setup - 0x300), -1 = disable
snd_irq - IRQ # for CS4232 chip (5,7,9,11,12,15)
snd_mpu_irq - IRQ # for MPU-401 UART (9,11,12,15)
snd_dma1 - first DMA # for CS4232 chip (0,1,3)
snd_dma1_size - max first DMA size in kB (4-64kB)
snd_dma2 - second DMA # for Yamaha CS4232 chip (0,1,3), -1 = disable
snd_dma2_size - max second DMA size in kB (4-64kB)
```

You would do a "modprobe snd−card−cs4231 snd_port=0x534 snd_mpu_port=−1 snd_irq=5 snd_dma1=0 snd_dma1_size=NN snd_dma2=1 snd_dma2_size=NN" to load the driver for a "standard configured" soundcard. (Without midi−support, see the note at Yamaha OPL−3). If you used different values in /etc/isapnp.conf, then you would use the values here also (Note: it can be wise to use your brains anyway ;)

Note that the "NN" values need to be supplied, only I do not know what would be reasonable values. I do not know if the dma size option is really required.

CS4232/CS4232A chips

According to the INSTALL file you need to supply the main port for this card. Note that with the driver for 3235/6/7/8/9 cards, the one below, I ended up supplying all information (except DMA−size), otherwise the driver did not work. So you may as well use the whole command line to insert the driver. If you initialized the card with the isapnp−tools, you can probably get info from the /etc/isapnp.conf file for the following values:

```
snd_port - port # for CS4232 chip (PnP setup - 0x534)
snd_cport - control port # for CS4232 chip (PnP setup - 0x120)
snd_mpu_port - port # for MPU-401 UART (PnP setup - 0x300), -1 = disable
snd_fm_port - FM port # for CS4232 chip (PnP setup - 0x388), -1 = disable
snd_jport - joystick port for CS4232 chip (PnP setup - 0x200), -1 =
disable
snd_irq - IRQ # for CS4232 chip (5,7,9,11,12,15)
snd_mpu_irq - IRQ # for MPU-401 UART (9,11,12,15)
snd_dma1 - first DMA # for CS4232 chip (0,1,3)
snd_dma1_size - max first DMA size in kB (4-64kB)
snd_dma2 - second DMA # for Yamaha CS4232 chip (0,1,3), -1 = disable
snd_dma2_size - max second DMA size in kB (4-64kB)
```

You would do a "modprobe snd−card−cs4232 snd_port=0x534 snd_cport=0x120 snd_mpu_port=−1 snd_fm_port=0x388 snd_jport=−1 snd_irq=5 snd_dma1=0 snd_dma1_size=NN snd_dma2=1 snd_dma2_size=NN" to load the driver for a "standard configured" soundcard. (Without midi−support, see the note at Yamaha OPL−3, and no joystick support). If you used different values in /etc/isapnp.conf, then you would use the values here also (Note: it can be wise to use your brains anyway ;)

Note that the "NN" values need to be supplied, only I do not know what would be reasonable values. I do not know if the dma size option is really required.

CS4235/CS4236/CS4236B/CS4237B/CS4238B/CS4239 chips

According to the INSTALL file you need to supply the main port and control ports for this card. Note that with a CS4237B card, I ended up supplying all information (except DMA−size), otherwise the driver did not work. So you may as well use the whole command line to insert the driver, and not only supply snd_port and snd_cport. If you initialized the card with the isapnp−tools, you can probably get info from the /etc/isapnp.conf file for the following values:

```
snd_port - port # for CS4232 chip (PnP setup - 0x534)
snd_cport - control port # for CS4232 chip (PnP setup - 0x120)
snd_mpu_port - port # for MPU-401 UART (PnP setup - 0x300), -1 = disable
snd_fm_port - FM port # for CS4232 chip (PnP setup - 0x388), -1 = disable
snd_jport - joystick port for CS4232 chip (PnP setup - 0x200), -1 = disable
snd_irq - IRQ # for CS4232 chip (5,7,9,11,12,15)
snd_mpu_irq - IRQ # for MPU-401 UART (9,11,12,15)
snd_dma1 - first DMA # for CS4232 chip (0,1,3)
snd_dma1_size - max first DMA size in kB (4-64kB)
snd_dma2 - second DMA # for Yamaha CS4232 chip (0,1,3), -1 = disable
snd_dma2_size - max second DMA size in kB (4-64kB)
```

You would do a "modprobe snd−card−cs4236 snd_port=0x534 snd_cport=0x120 snd_mpu_port=−1 snd_fm_port=0x388 snd_jport=−1 snd_irq=5 snd_dma1=0 snd_dma1_size=NN snd_dma2=1 snd_dma2_size=NN" to load the driver. (Without midi−support, see the note at Yamaha OPL−3, and no joystick support). Notes:

* the "NN" values need to be supplied, only I do not know what would be reasonable values.
* my CS4237B works fine without explicit dma size option.

5.4 The kerneld approach

`kerneld` is a daemon that inserts modules on request, and unloads them once they are not in use anymore. Since I have no experience with kerneld, I do not know if the information below is accurate. The info comes from the INSTALL file in the ALSA−drivers package. Excellent information about kerneld can be found in the kerneld−mini−HOWTO.

Follow these steps:

* Edit your /etc/conf.modules (see below for examples)
* Run 'modprobe snd−card' where card is name of your card [Which I find rather strange, since kerneld is supposed to load them? VS]

Example for /etc/conf.modules for Gravis UltraSound PnP soundcard:

```
alias char-major-14 snd
alias snd-minor-oss-0 snd-interwave
alias snd-minor-oss-3 snd-pcm1-oss
alias snd-minor-oss-4 snd-pcm1-oss
alias snd-minor-oss-5 snd-pcm1-oss
alias snd-minor-oss-12 snd-pcm1-oss
alias snd-card-0 snd-interwave
options snd snd_major=14 snd_cards_limit=1
options snd-interwave snd_index=1 snd_id="guspnp" snd_port=0x220 snd_irq=5
snd_dma1=5 snd_dma2=6
```

Example if you want use more soundcards in one machine (configuration below is for Sound Blaster 16 and Gravis UltraSound Classic):

```
alias char-major-14 snd
alias snd-minor-oss-0 snd-mixer
alias snd-minor-oss-3 snd-pcm1-oss
alias snd-minor-oss-4 snd-pcm1-oss
alias snd-minor-oss-5 snd-pcm1-oss
alias snd-minor-oss-12 snd-pcm1-oss
alias snd-card-0 snd-sb16
alias snd-card-1 snd-gusclassic
options snd snd_major=14 snd_cards_limit=2
options snd-sb16 snd_index=1 snd_port=0x220 snd_irq=5 snd_dma8=1 snd_dma16=5
options snd-gusclassic snd_index=2 snd_irq=11 snd_dma1=6 snd_dma2=7
```

Example if two Gravis UltraSound Classic soundcards are present in system:

```
alias char-major-14 snd
alias snd-minor-oss-0 snd-mixer
alias snd-minor-oss-3 snd-pcm1-oss
alias snd-minor-oss-4 snd-pcm1-oss
alias snd-minor-oss-5 snd-pcm1-oss
alias snd-minor-oss-12 snd-pcm1-oss
alias snd-card-0 snd-gusclassic
alias snd-card-1 snd-gusclassic
options snd snd_major=14 snd_cards_limit=2
options snd-gusclassic snd_index=1,2 snd_port=0x220,0x260 snd_irq=5,11
snd_dma1=5,6 snd_dma2=7,3
```

5.5 Backwards compatibility

If you want to preserve OSS/Free or OSS/Linux compatibility, you need to insert one more driver: the snd−pcm1−oss driver for OSS−compatibility. Issue a

```
modprobe snd-pcm1-oss
```

This will give you /dev/audio and /dev/dsp−support, just as the OSS/Free (kernel) drivers and OSS/Linux (the $25 ones) do. Note that this is only an emulation.

---

6. Testing and using

Now you should test if the sound driver really is available, then try to use it.

6.1 The /proc filesystem

You can find a lot of useful information about your system in the /proc subdirectory. /proc is a "virtual" filesystem, meaning that it does not exist in real life, but merely is a mapping to various processes and tasks in your computer. In order for /proc to work, you need to have support for it compiled into your kernel. Most linux distributions have this as a default, but if you compiled a kernel and left /proc out obviously there won't be anything in /proc.

/proc/modules gives information about loaded modules. Once the ALSA sound drivers are loaded, if you type cat /proc/modules you should see something like:

```
snd-pcm1-oss      4            0
snd-sb16          1            1
snd-sb-dsp        4    [snd-sb16]      0
snd-pcm1          4    [snd-pcm1-oss snd-sb-dsp]      0
snd-pcm           3    [snd-pcm1-oss snd-sb16 snd-sb-dsp snd-pcm1]    0
snd-mixer         3    [snd-pcm1-oss snd-sb16 snd-sb-dsp]      1
snd-mpu401-uart   1    [snd-sb16]      0
snd-midi          4    [snd-sb16 snd-sb-dsp snd-mpu401-uart]   0
snd-opl3          1    [snd-sb16]      0
snd-synth         1    [snd-sb16 snd-opl3]     0
snd-timer         1    [snd-opl3]      0
snd               8    [snd-pcm1-oss snd-sb16 snd-sb-dsp snd-pcm1 snd-pcm snd-mixer snd-mpu401-uart snd-midi snd-opl3 snd-synth snd-timer]    0
```

If something went wrong during the installation of the driver, you will still see a couple of "snd" devices, but there won't be sound support.

For example (Note: you should never issue this command as follows, the cs4236 driver needs options):

```
win3:~# modprobe snd-card-cs4236
/lib/modules/2.0.35/misc/snd-card-cs4236.o: init_module: Device or resource busy
snd-mixer: Device or resource busy
win3:~# cat /proc/modules
snd-cs4236        2           0
snd-cs4231        3    [snd-cs4236]    0
snd-timer         1    [snd-cs4231]    0
snd-pcm1          4    [snd-cs4236 snd-cs4231] 0
snd-mixer         3    [snd-cs4236 snd-cs4231] 0
```

```
snd-pcm            3    [snd-cs4236 snd-cs4231 snd-pcm1]       0
snd-mpu401-uart    1          0
snd-midi           4    [snd-mpu401-uart]       0
snd-opl3           1          0
snd-synth          1    [snd-opl3]      0
snd-timer          1    [snd-cs4231 snd-opl3]   0
snd                8    [snd-cs4231 snd-timer snd-pcm1 snd-mixer snd-pcm]     0
```

You can check the existence of a soundcard by looking in /proc/asound/cards. For example:

```
bash$ cat /proc/asound/cards
0 [card1          : SB16 - Sound Blaster 16
                    Sound Blaster 16 at 0x220, irq 5, dma 1&5
```

In the previous example (where I forgot the options) the output would have been:

```
win3:~# cat /proc/asound/cards
--- no soundcards ---
```

A working CS4236 card would produce

```
0 [card1         ]: CS4236 - CS4237B
                    CS4237B at 0x534, irq 7, dma 1&0
```

If you checked and doublechecked your settings and still see no sound card, take a look at the troubleshooting section.

The `/proc/asound/` virtual directory shows lots of other information about the driver. Please note that /proc/asound/ will only exist after you inserted the first ALSA module. If there is no /proc/asound, it simply means that the "snd" module was not loaded properly. You can find installed cards in /proc/asound/cards, then find information about card0 in /proc/asound/0, /proc/asound/1 for card1 etcetera.

If `cat /proc/asound/card1/pcm0` shows something like

```
ES1370 DAC2/ADC
Playback isn't active.
Record isn't active.
```

this means that your driver is ready to go, but is not doing anything right now. (So everything went well).

For users of a 2.0.x kernel there is a third method to find information about the sound devices, namely if you inserted the OSS compatible driver there is a /dev/sndstat device. The ALSA drivers kindly request that you not to rely on this information as it is only there for compatibility with the OSS drivers and better information can easily be obtained from /proc/asound/. In kernel 2.2.x ALSA uses the kernel soundcore and therefor cannot emulate /dev/sndstat, since it would interfere with the OSS drivers.

6.2 The mixer

Once the drivers for your sound card have been installed and your /proc filesystem tells you so, you can try to make a real sound. To do this, you need to set the mixer volumes to a reasonable value. You need the ``amixer'' from the alsa−utils package for this. First of all, install the utility package, or at least put the "amixer" command in some reasonable place (like /usr/local/bin).

Version 0.3.2 and later have an interface that differs from the OSS drivers. If you type just ``amixer'' you will see the mixer elements and their value. One of these elements could be ``Master volume'' for example, and could look like:

```
Group 'Master',0
  Capabilities: volume
  Channels: Front-Left Front-Right
  Limits: min = 0, max = 31
  Front-Left: 31 [100%] [on] [---]
  Front-Right: 26 [84%] [on] [---]
```

Unfortunately, I do not know how to set left and right volumes independently. With amixer, you can change volumes with the ``amixer set'' command. For example, to change the Master volume, you would issue a

```
amixer set Master 15
```

Please note that the names of the elements can be different for different types of sound cards. Also note that amixer is case dependent, so ``amixer set masteR 10'' will not work. For more information, please look in the amixer man page.

If you have a 0.3.0−pre4 ALSA, then amixer works just like normal mixer programs. You can look at the mixer settings by typing ``amixer''. This command lists the ``mixer settings'', or as you would normally call it, the volume settings of the various parts of the soundcard. The output from amixer can greatly differ from card to card. My Soundblaster 16 shows:

```
Master          0  % (-14.00dB) : 0  % (-14.00dB)
Bass            0  % (-14.00dB) : 0  % (-14.00dB)
Treble          0  % (-14.00dB) : 0  % (-14.00dB)
Synth           0  % (-62.00dB) : 0  % (-62.00dB)
PCM             0  % (-62.00dB) : 0  % (-62.00dB)
Line-In         0  % (-62.00dB) : 0  % (-62.00dB) Mute
MIC             0  % (-62.00dB) : 0  % (-62.00dB) Mute
CD              0  % (-62.00dB) : 0  % (-62.00dB) Mute
In-Gain         0  % (-18.00dB) : 0  % (-18.00dB)
Out-Gain        0  % (-18.00dB) : 0  % (-18.00dB)
PC Speaker      0  % (-18.00dB) : 0  % (-18.00dB)
```

If you only get a message like ``amixer: Specify command...'', then you are using the ALSA 3.2 utilities. I suggest you to upgrade to 0.4.1e or later, or to go back to 0.3.0−pre4.

Mixer settings for playing

You have noticed the "Mute" entry for some devices. This means that this particular device will be zeroed out, whatever volume setting you use. Some cards (the CS4237B in the example) even mute their master channel. So, for the CS4237B, I would have to type

```
amixer set "Master d" unmute
```

to even be able to produce any sound at all. The Soundblaster does not have muted output, but

```
amixer set Master 100 unmute
```

would set the volume to 100% − and unmute it if it would have been muted. You can use a number, a word like "mute" or "unmute", or both. Type

```
amixer set "Master d" 100; amixer set PCM 100 unmute
```

to set the CS4237B card to maximum master volume and unmute PCM volume and set it to maximum.

If you use an older version of amixer, you need to leave out the ``set'' part of the command, so you would just type

```
amixer "master d" 100
```

Mixer parts

The various mixer parts may confuse you if you have no knowledge of digital sound production. The sound−HOWTO may help a bit, but a very short introduction is here.

You will probably only need few mixer elements: one of them is the ``CD'' setting (this is analog sound of your CD player, most CD players are connected with a 3 or 4 wire red/white/black cable).

The ``PCM'' setting is used for most applications. Programs like mpg123, xmms, speakfreely, realplayer and most others use the PCM channel.

``MIC'' stands for microphone, ``line−in'' is an (optional) extra input at the back of your sound card.

The various ``gain'' parts offer extra amplification for various uses and are pretty self−explanatory. (Like: record−gain is extra amplification for the recording channel, which can be useful if you use a microphone).

Mixer settings for recording

You would set the CD channel to record by typing

```
amixer set CD capture
```

and stop the recording setting again by typing

`amixer set CD nocapture`.

Note that older amixer programs use ``amixer CD rec'' and ``amixer CD norec'' for this.

If you would like to record something from the microphone, you would probably use

`amixer set "Input Gain" 100; amixer set Mic 100 capture mute`.

(Using the microphone input unmuted will produce loud high−pitched sound if your mic picks up its own signal from the speakers again). Most microphones have a ``gain'' setting to boost the microphone volume; you are most likely going to need it to pick up any sound from the microphone at all.

Again, older amixer programs use ``amixer "input gain" 100; amixer mic 100 rec mute''.

Other mixer settings

Unfortunately I have not been able to change the volume of the "3d center" and "3d space" settings with amixer 0.3.0−pre4. I haven't tried yet with 0.4.1e (this particular machine is still running 2.0.38). If anyone succeeds please let me know. I can use alsamixer for this job, but alsamixer was not ported to the 0.4.1e version yet.

The ALSA FAQ says that it is possible to restore mixer settings with `cat <file> > /proc/asound/#/mixerC0D0`, where <file> was obtained from /proc/asound/#/mixerC0D0. I have not been able to reproduce this as my system complains about non−existing devices. Then there is the ``alsactl'' program, which I don't use. I invite you (yes, you!) to write this section.

6.3 The /dev/snd/ devices

The alsa drivers have native sound−devices in the /dev/snd/ directory. If you have one card you might see the following devices:

```
/dev/snd/pcmC0D0 − the raw audio device for the card
/dev/snd/mixerC0D0 − the mixer for card 0
/dev/snd/controlC0D0 − the control device for card 0
```

The first number means the number of the soundcard, the second number (if any) is the number of the device. A sound card with two PCM devices would have a pcmC0D0 and pcmC0D1 device. Please note: the ALSA devices have changed between the previous version. Older ALSA drivers use /dev/snd/pcm00 (first number is the card, second number is the device). If this HOWTO uses the older notation, please drop me a line so I can correct it.

Now you are ready to put any soundfile you want into the PCM device of the first card. So try to cat any textfile (any file) to /dev/snd/pcmC0D0, like this: `cat <filename> > /dev/snd/pcmC0D0`. The filename can be any file, as long as it has some length. If you have a soundfile lying around somewhere, you could try that. You could also get the file at http://www.ldp.org/sounds/english.au this is Linus Torvalds saying how to pronounce Linux.

The default setting of your sound device is 8000 Hz, 8 bit. That means that the "english.au" file mentioned above will produce speech, other test files will probably just produce noise. If you do not hear anything, check your speakers, try to run "amixer" again or consult a doctor. (Later on you can easily use the full 48 KHz, 16 bit features of your sound card, by using your favourite sound player like sox or mpg123).

If you loaded the ``snd−pcm1−oss'' module, you can also use the OSS−compatibility to access your sound card. The following mappings are made:

```
/dev/snd/pcmC0D0 -> /dev/audio0 (/dev/audio) -> minor 4
/dev/snd/pcmC0D0 -> /dev/dsp0 (/dev/dsp) -> minor 3
/dev/snd/pcmC0D1 -> /dev/adsp0 (/dev/adsp) -> minor 12
/dev/snd/pcmC1D0 -> /dev/audio1 -> minor 4+16 = 20
/dev/snd/pcmC1D0 -> /dev/dsp1 -> minor 3+16 = 19
/dev/snd/pcmC1D1 -> /dev/adsp1 -> minor 12+16 = 28
/dev/snd/pcmC2D0 -> /dev/audio2 -> minor 4+32 = 36
/dev/snd/pcmC2D0 -> /dev/dsp2 -> minor 3+32 = 35
/dev/snd/pcmC2D1 -> /dev/adsp2 -> minor 12+32 = 44
```

You probaly want to use the ``snd−mixer−oss'' module as wel, so you can use the backwards compatible mixer.

6.4 Additional information

The INSTALL file in the ALSA driver directory mentions some tricks to tell the driver which settings to use. If you need these commands it will depend on the application you use to play sound. Regular sound playing applications, like mpg123, sox (mostly called with the ``play'' command), or X11 applications like RealPlayer will probably do fine without these. I never used these anyway.

/proc/asound/#/pcm#0

```
"Playback erase" - erase all additional informations about OSS applications
"Playback <app_name> <fragments> <fragment_size> [<options>]"
"Record erase" - erase all additional informations about OSS applications
"Record <app_name> <fragments> <fragment_size> [<options>]"
```

`<app_name>` – name of application with (highter priority) or without path

`<fragments>` – number of fragments or zero if auto

`<fragment_size>` – size of fragment in bytes or zero if auto

`<options>` – optional parameters

`WR_ONLY` – if application tries open pcm device with O_RDWR driver rewrites this to O_WRONLY (playback) – good for Quake etc...

Examples:

```
echo "Playback x11amp 128 16384" > /proc/asound/0/pcm0o
echo "Playback squake 0 0 WR_ONLY" > /proc/asound/0/pcm0o
```

/proc/asound/#card#/sb16

```
"Playback 8" -> driver will use always 8-bit DMA channel for playback.
"Playback 16" -> driver will use always 16-bit DMA channel for playback.
"Playback auto" (default) -> driver will use auto mode (first opened direction will use 16-bit DMA channel).
"Record 8" -> driver will use always 8-bit DMA channel for record.
"Record 16" -> driver will use always 16-bit DMA channel for record.
"Record auto" (default) -> driver will use auto mode (first opened direction will use 16-bit DMA channel).
```

Example: `echo "Record 16" > /proc/asound/0/sb16`

For further reference, please consult the INSTALL file.

---

7. Tips and Troubleshooting

Please take a look at the FAQ file in the sound driver directory. This section is still under construction.

7.1 Compiling the driver

Linux kernel sourcetree

If your ALSA drivers do not compile correctly and tell you things about ``version.h'' or other header−files that cannot be found, this can mean that you do not have the kernel header files. Take a look at the kernel−HOWTO, unpack a recent kernel in /usr/src and issue a `make config`.

Cannot create executables

The utils also contain code written in c++. Most of us have a c++ compiler either from gcc or egcs but make sure you also have the libstdc++−devel package installed, else when you run the configure script for the utils, your system will stump you with an error message saying your ``c++ compiler cannot create executables''.

7.2 Loading the driver

Please check the following items.

Sound devices

ALSA uses special devices in the /dev−tree. Make sure you have run the ./snddevices script in the alsa−drivers source directory.

Sound card compatibility

Are you 100% sure that your sound card IS supported ? Do check it again. Sometimes an X123 is not exactly an X123b and you might be wasting time. On the other hand, even a supported card can give you troubles − it took me two hours to figure out the installation of a CS4237B which was, after all, just a fine example of RTFM.

``Device busy'' or ``unresolved symbols''

You might have a 2.0.x kernel with sound support compiled in, or the OSS/Lite (kernel) sound driver could be loaded (check with `cat /proc/modules`). Remove the driver or recompile the kernel (have a look at the Kernel−HOWTO).

The sound module in the 2.0 series kernel is called ``sound.o'' and should **not** be active. (The ALSA driver ``snd.o'' is OK, though).

If you have a 2.2.x series kernel without sound driver compiled in, the ALSA drivers will not work, too.

2.0 kernels

I know it this is confusing, so let me try to explain it one more time. If you have a **2.0.x series kernel** (the command ``uname −a'' tells you something like ``Linux penguin 2.0.35 #6 Wed Sep 23 10:19:16 CEST 1998 i686 unknown'') then you need to **leave out** sound drivers in the kernel. **ALSA 0.4.x and later do not work with the 2.0 series kernel**

2.2 kernels

If you have a **2.2.x series kernel** you do **need the sound drivers**. A 2.2 series kernel should be compiled **with sound support, but without any sound card driver**. So you select sound support but make sure that no specific sound card driver will be compiled.

References to other drivers

Another reason why the driver complains that the device is busy could be that the file /etc/conf.modules still has references to the soundcard drivers. You should delete these and leave only the references to the ALSA−driver. (If there are other non−sound−related drivers there, then you can probably leave these as−is).

Unresolved symbols revisited

Another source of ``unresolved symbols'' messages could be a new kernel with older drivers. Please recompile the ALSA drivers after you recompile a new kernel. This will make sure that the drivers match your new kernel.

Check the PnP setup

Are you sure that your card is active? Take another look at the PnP−HOWTO and check if you activated your sound card correctly.

Are your parameters right ?

Check, doublecheck your sound card parameters. Please note: 534 is not 543, nor is 0x534 the same as 534.

Also, some sound cards must be loaded by a different name than might be expected. Take a break, a beer or whatever, and look again at your ``modprobe'' command. For example the Crystal 4232 driver should be inserted by modprobe snd−card−cs4232, not ``snd−cs4231'', and the SoundBlaster PCI 64 should be loaded with ``snd−card−audiopci'', not snd−es1370. (It's all in the docs, and even though I wrote the HOWTO, I once spent an evening trying to persuade snd−cs4231 to make sound).

7.3 Driver loaded... but no (or hardly any) sound

Unmuting

The ALSA drivers can use the ``muting'' facilities that most soundcards have. If you loaded the sound drivers and everything is fine but you get nothing but silence, then you probably forgot to unmute your card. You need ``amixer'' or ``alsamixer'' for this, both from the ALSA−util package. Just typing

```
amixer set -c 1 Master 70 unmute
amixer set -c 1 PCM 70 unmute
amixer set -c 1 CD 70 unmute
```

should do for most applications. Please note that for the older amixer command you need to leave out the ``set'' in the command line.

Gain

Most sound cards have a separate mixer part for extra input or output boosting. This entry is most likely called the ``gain'', ``in−gain'' for input and ``out−gain'' for output. Setting this gain to an appropriate level will greatly help you getting the maximum volume out of your speakers (think about your parents/neighbours/ears though). So a command like

```
amixer set out-gain 100 unmute
```

will probably help.

OSS/Linux compatibility

If this is the first time you use the ALSA drivers and you used the built−in sound drivers before, you probably want to have backwards compatible sound (i.e. use the /dev/pcmX devices). You need to load the ``OSS compatibility driver'' for this. Do a `modprobe snd-pcm1-oss && modprobe snd-mixer-oss`. (See the end of the section about loading the driver). Please note: snd−pcm1−oss is not equal to snd−pcm1, you **need** snd−pcm1−oss for old−fashioned sound support and snd−mixer−oss for (you guessed it) the mixer.

Cannot open mixer

If you have tried to install a couple of different ALSA versions, then sometimes the mixer cannot be opened anymore. This happens if you have tried 0.3.2 and want to downgrade to 0.3.0−pre4 (IIRC). You should delete all libasound files and links from /usr/lib and then recompile libraries and utils:

```
rm /usr/lib/libasound.*
```

Just to be safe, remove all ALSA sound drivers afterwards, then recompile and install and reload the drivers.

7.4 General suggestions

Try using ``insmod''

It can always be useful to start with "insmod" instead of kerneld. Maybe you actually see the error on screen.

Read the INSTALL file.

A lot of information can be found in the INSTALL file in the drivers directory. If your driver won't work check if there is additional information available.

Debug messages

As a last resort, you can rebuild the driver and tell it to send debug information to /var/log/messages. Go to the driver−directory with `cd /usr/src/alsa-driver-....` and type:

```
./configure --with-debug=detect; make clean; make
```

Remove the driver (as far as it is active, see below for a general remove statement). Then use the "modprobe" statement you used before to insert the newly compiled driver. Look in /var/log/messages if there are any messages.

If all else fails...

If these messages doesn't help you, send a message to the ALSA users mailing list, alsa−user@alsa−project.org.

Include the following information:

- soundcard name + chip names present on your soundcard
- relevant sections in your isapnp.conf if you have ISA PnP soundcard
- your conf.modules or line which you activate ALSA driver
- all messages from /var/log/messages which should be relevant to the ALSA driver

7.5 Bug reports

If you found a bug, the ALSA developers would like to know the following things (at minimum)

1. driver + kernel version: 'cat /proc/asound/version'
2. soundcard info
    - soundcard name provided by manufacture
    - list of chips which soundcard have onboard
    - contents of 'cat /proc/asound/cards'
3. all messages from /var/log/messages which should be relevant to ALSA driver
4. problem description

7.6 Tip: playing CD's

If you use kmod/kerneld and the ALSA drivers to play CD's, then kmod/kerneld probably do not load the drivers as expected. This is due to the fact that a command line CD player only tells the CD player to start playing without using any of the devices that tell kmod/kerneld that there is sound to occur. Using modprobe may be your only solution to this problem.

7.7 Tip: installing the MIDI serial driver

Normally, the IO port of the serial device is owned by the standard serial device driver. So before you can do ``modprobe snd−serial'' we have to tell the driver to release the serial device.

Here is the procedure.

```
setserial /dev/ttyS0 uart none
modprobe snd-serial
```

(Replace /dev/ttyS0 with the appropriate /dev/ttySx device if your MIDI device uses a different serial device).

7.8 Tip: new kernel? New modules!

After you upgrade your kernel, you probably need to recompile the ALSA drivers. If they are still in the original /usr/src directory, then please do not forget to issue a `make clean` before you do the ./configure, make, make install thing.

Oh, and then there is this anomaly in kernel numbering: a ``2.2.0ac1'' kernel that is ``not a number'' − says the configure script. I think this was resolved in newer scripts, otherwise you should maybe change the kernel version in the source.

7.9 Tip: KDE and ALSA drivers

Suppose you have KDE up and running but you cannot get system sounds to work, like for opening windows, changing desktops, etc. Sound works in general. If your cd player and mp3 player and mixer all do work, then it's probably just "kwmsound" that's lacking.

So: make sure "kwmsound" is in your startscript ($KDEDIR/bin/startkde)

7.10 Tip: use the ALSA devices

If you had sound support in your Linux before, then your applications will probably all point to /dev/pcm0, /dev/audio and /dev/mixer. This is fine, if you use OSS compatibility with the snd−pcm1−oss module. It might be better, however, to use the real ALSA devices, those found in /dev/snd/.

7.11 Tip: removing all modules

Removing 10+ modules one by one is not the way to go. Luckily, all modules start with the "snd−" prefix, so a little command line programming will do. You can easily remove ALSA sound by issuing a command like:

```
cat /proc/modules|gawk '/^snd-/{print $1}'|xargs -i rmmod {}
```

Juergen Kahrs wrote: ``I have a script that also removes soundcore and soundlow and sound if present and if they are not in use. This script processes /proc/modules three times so there should not be too many modules left after processing''. His solution is

```
awk '/^snd/||/^sound/&&($3==0){system("rmmod " $1)}' /proc/modules /proc/modules /proc/modules
```

Please note: if some module is dependent on another module you cannot just remove the "higher" one. This means that you might need to issue a second removal statement. (I never encountered this situation though, it seems that you can remove the ALSA modules in the order they appear in /proc/modules).